REMARKS/ARGUMENTS

Applicants acknowledge the rejection of claims 1, 2, 4-10, 12, and 14-23 with a right to traverse. Claims 1, 4, 7-10, 12, 14, 15, 18, 19, 22, and 23 are currently amended to clarify the claimed language. Claims 3, 11, and 13 are canceled without prejudice. As a result, claims 1, 2, 4-10, 12, and 14-23 are now pending in this application. Applicants respectfully request further examination and reconsideration of the rejections for the reasons stated below.

Claim Rejection – 35 USC 112

Applicants have deleted the word "substantially" from independent claims 1, 10, and 12 to obviate the 35 USC 112 rejections.

Claim Rejections - 35 U.S.C. §102

Independent claims 1, 10 and 12 were rejected as being allegedly anticipated by U.S. Patent No. 5,371,878 (hereinafter "Coker"). Applicants respectfully traverse in view of the following.

Currently amended independent claim 1 recites a boot method for synchronizing a microcontroller and a virtual microcontroller of an In-Circuit Emulation system in lock step through executing a set of boot code in the microcontroller and executing a dummy code in the virtual microcontroller, wherein the dummy code is a set of timing code to

take the same number of clock cycles as the microcontroller and wherein the set of boot code is inaccessible to the virtual microcontroller.

First, Coker does not disclose how the synchronization of the target-ECS and the shadow system in-lock step is realized through the booting process. Although Coker discloses the shadow system operating in sync with the target-ECS to provide the ability to recreate and debug, Applicants respectfully disagree with the notion that Coker teaches how the synchronization of the target-ECS and the shadow system is realized through the booting process of the In-Circuit Emulation system. Instead, Coker merely mentions how the shadow system executes the same or similar software as the target-ECS from system start-up or reset. (See Coker, col. 2, lines 56-59).

Second, the rejection states that "Coker specifically discloses that the shadow system central processing unit performs numerical operations to send outputs to specific locations within its RAM rather than to complex output registers (in contrast to the target-ECS software)." Coker further teaches that the shadow system maintains an execution state at any given time corresponding to a known execution state in the target-ECS and thus produces a mirror-image of the target-ECS through performing the numerical operations on input data with the same value, memory location and relative timing as the target-ECS. (See Coker, col. 3, lines 6-12).

The disclosure by Coker implies that the shadow system executes software similar to the one executed by the target-ECS, if it is not the same, on the same data processed by the target-ECS to simulate the target-ECS. Coker further discloses that "the shadow system receives its input data from the input registers of the target-ECS and stores the input data in its RAM." (See Coker, col. 2, lines 61-63). This is readily evident when Coker further elaborates that "[i]f the shadow system is ready to receive data (i.e., sent

from the target-ECS) and no data is currently stored in the data buffer, the shadow system halts execution of instructions and waits until the next unique input event is available from the data buffer." (See Coker, col. 4, lines 31-35). "The shadow system uses only the input events (not output events) received by the target-ECS and the corresponding execution states of the shadow system to exactly re-create all the execution states of the target-ECS." (See Coker, col. 9, lines 58-62).

Accordingly, the input data are required by the shadow system according to Coker because the main objective of the shadow system is to mirror the execution state of the target-ECS, thus making the shadow system available for recreating and debugging of the target-ECS. Thus, it is apparent that the software executed by the shadow system should be at least similar to the one performed by the target-ECS.

In contrast, the virtual microcontroller of the embodiment, unlike the shadow system of Coker, does not require any input data from the microcontroller at least during the booting process of the microcontroller. Data are copied only after the booting process of the microcontroller is completed. This is possible because the virtual microcontroller does not execute any meaningful numerical operations during the booting process of the microcontroller. The virtual microcontroller executes a set of timing code to take the same number of clock cycles as the microcontroller uses to execute the set of boot code, thus achieving the synchronization of the microcontroller and the virtual microcontroller in lock-step (with the copying of registers and memory from the microcontroller to the virtual microcontroller towards the end of the synchronization) without processing any input data to the microcontroller.

To further emphasize this distinction, "a dummy code" is inserted to the claim language of independent claims 1, 10, and 12. Applicants respectfully assert that no new

matter has been added as a result of these claim amendments, and that these claim amendments are adequately supported by the application and figures as originally filed. The virtual microcontroller according to the amended claims executes the dummy code to use up the same number of clock cycles as the microcontroller uses to execute the set of boot code during the booting process. Unlike the software used by the shadow system of Coker, the software executed by the virtual microcontroller during the booting process of the microcontroller is dummy code. As independent claim 1 states, the dummy code is employed to use up the clock to achieve the lock step synchronization of the microcontroller and the virtual microcontroller during the booting process of the microcontroller.

In contrast to the shadow system of Coker, the virtual microcontroller does not require input data from the microcontroller at least during the booting process of the microcontroller. Because no input from the microcontroller is fed to the virtual microcontroller during the booting process of the microcontroller, the set of boot code used by the microcontroller is kept from the virtual microcontroller. This may be necessary to protect any proprietary information (e.g., passwords, algorithm, etc.) that is stored within the microcontroller. In contrast, assuming arguendo that the software used by the shadow system in Coker is not the same as that of the target-ECS, it still has to perform meaningful operations to input data with the same value, memory location and relative timing of the target-ECS to produce a mirror-image of the target-ECS. Thus, the software used by the shadow system cannot be a dummy code as claimed in independent claim 1.

Accordingly, Coker fails to teach or suggest the recited limitations of independent claim 1. Firstly, Coker does not disclose a booting method to synchronize the In-Circuit

Emulation system. Assuming arguendo the booting method was inherent in Coker, Coker fails to anticipate each and every element as set forth in independent claim 1. That is, Coker discloses the shadow system executing meaningful software code (unlike the dummy code recited in independent claim 1) to mirror the operation of the target-ECS.

Therefore, Applicants respectfully request the withdrawal of the rejection under 35 USC 102. Independent claims 10 and 12 recite limitations similar to that of independent claim 1 and are therefore patentable over the cited reference for the same reasons. As such, allowance of independent claims 1, 10, and 12 is earnestly solicited. Additionally, allowance of the rest of the claims that depend on the independent claims is earnestly solicited.

Claim Rejections - 35 U.S.C. §103

Applicants respectfully acknowledge the withdrawal of the Tzori reference (U.S. Patent No. 6,202,044) with the amendments to the claim language, particularly in response to the limitation that "at least one portion of said set of timing code is different from said set of boot code." Claims 1-2, 4-10, 12 and 14-20 are currently rejected as allegedly being unpatentable over U.S. Patent No. 5,371,878 (hereafter "Coker") in view of "How Debuggers Work" by Jonathan B. Rosenberg (hereafter "Rosenberg"). Applicants respectfully traverse in view of the following.

Currently amended independent claim 1 recites a boot method for synchronizing a microcontroller and a virtual microcontroller of an In-Circuit Emulation system in lock step through executing a set of boot code in the microcontroller and executing a dummy code in the virtual microcontroller, wherein the dummy code is a set of timing code to

take the same number of clock cycles as the microcontroller and wherein the set of boot code is inaccessible to the virtual microcontroller.

As Applicants stated earlier, Coker does not anticipate independent claim 1 as it does not disclose how to synchronize the shadow system and the target-ECS in lock-step during the booting process, nor does Rosenberg teach or suggest the synchronization of the In-Circuit Emulation system using the booting process.

Moreover, even if the teaching of the synchronization of the shadow system and the target-ECS was inherent for the sake of argument, Coker does not disclose at least two elements of the independent claim, e.g., the software used by the virtual microcontroller being a dummy code, thus the inaccessibility of the booting code of the microcontroller to the virtual microcontroller. Additionally, Rosenberg neither suggests nor teaches the two elements during the booting process of the In-Circuit Emulation system.

Therefore, the above referenced rejection fails to establish a prima facie evidence in support of the rejection. Accordingly, independent claim 1 is not rendered obvious under 35 USC 103 over Coker in view of Rosenberg. Independent claims 10 and 12 recite limitations similar to that of independent claim 1 and are therefore patentable for the same reasons. Dependent claims 2, 4-9, and 14-23 are patentable by virtue of their dependency to the independent claims.

CONCLUSION

In light of the above listed remarks, reconsideration of the rejected Claims is requested. Based on the arguments presented above, it is respectfully submitted that Claims 1-2, 4-10, 12 and 14-23 overcome the rejections of record and, therefore, allowance of Claims 1-2, 4-10, 12 and 14-23 is earnestly solicited.

Please charge any additional fees or apply any credits to our PTO deposit account number: 23-0085

Respectfully submitted,

Murabito, Hao & Barnes

Date 7/02/2007 By

Steve S. Ko
Reg. No. 58,757

Two North Market Street
Third Floor
San Jose, CA 95113
(408) 938-9060